Downsampled Rendering

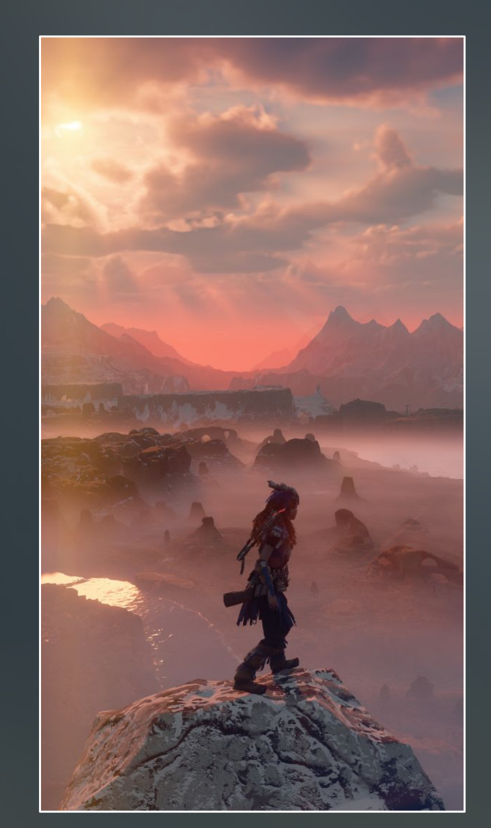
Ray-marching a 1:1 volumetric atmosphere interactively

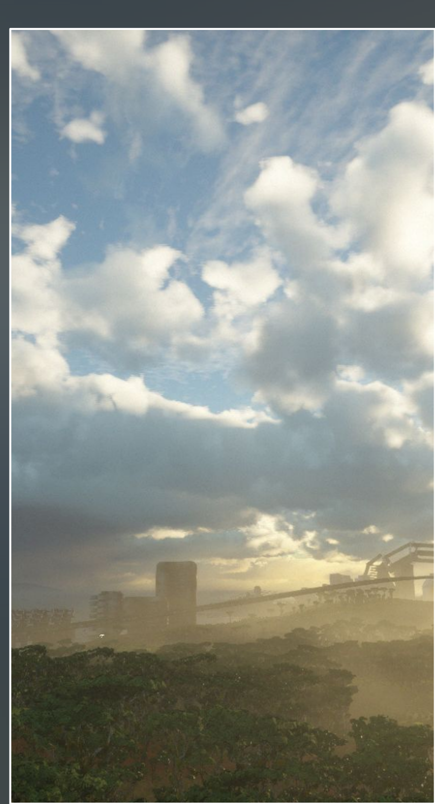
(see slide notes for narration)

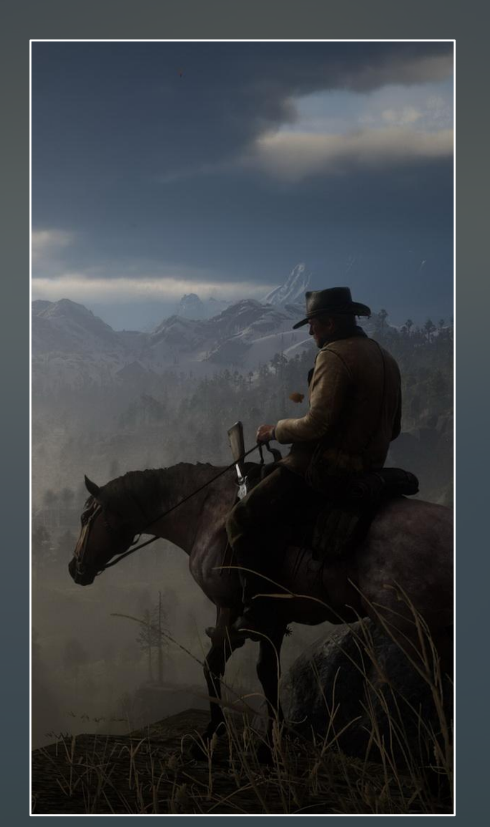

Motivation
- 1:1 scale terrain engine
- atmospheric scattering [Bruneton2008]
- missing cloudscape
- missing terrain shadows on atmosphere
→ Volumetric atmosphere simulation

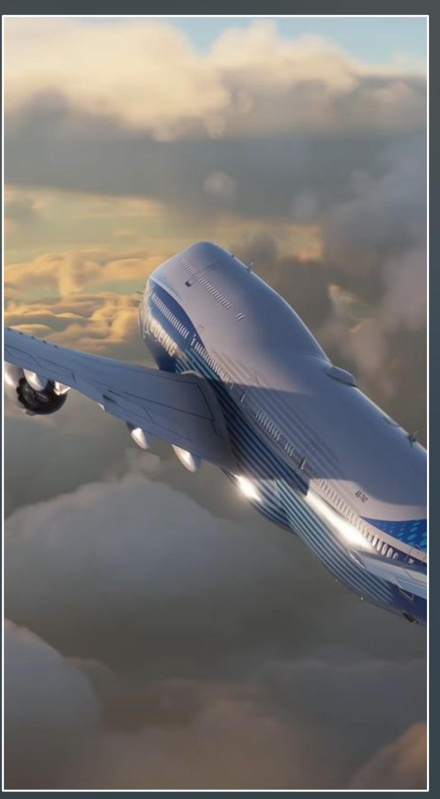

Motivation

Motivation









2015 2016 2018 2019



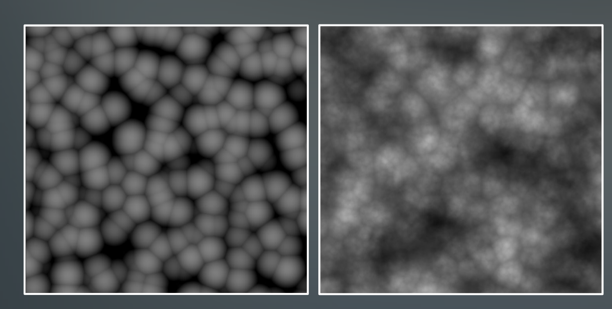
Volumetric clouds

Overview

1. Modeling
2. Lighting
3. Rendering

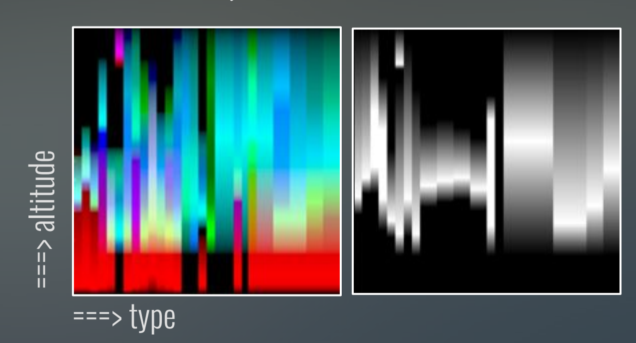
Modeling

- Local cloud shape 3D textures

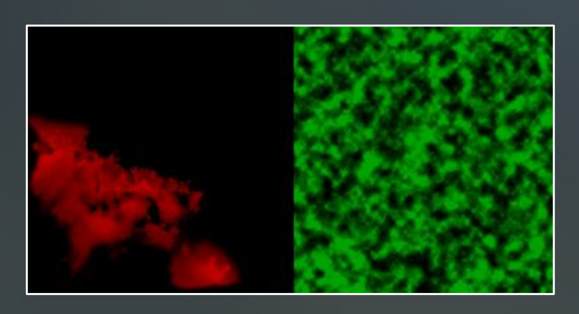


Layered Worley noise

- Density modifiers



- Global 2D cloud coverage maps

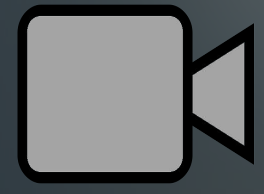


- Wind animation
  -> simple sampling offset

[Schneider2015]
[Hillaire2016]
[Bauer2019]



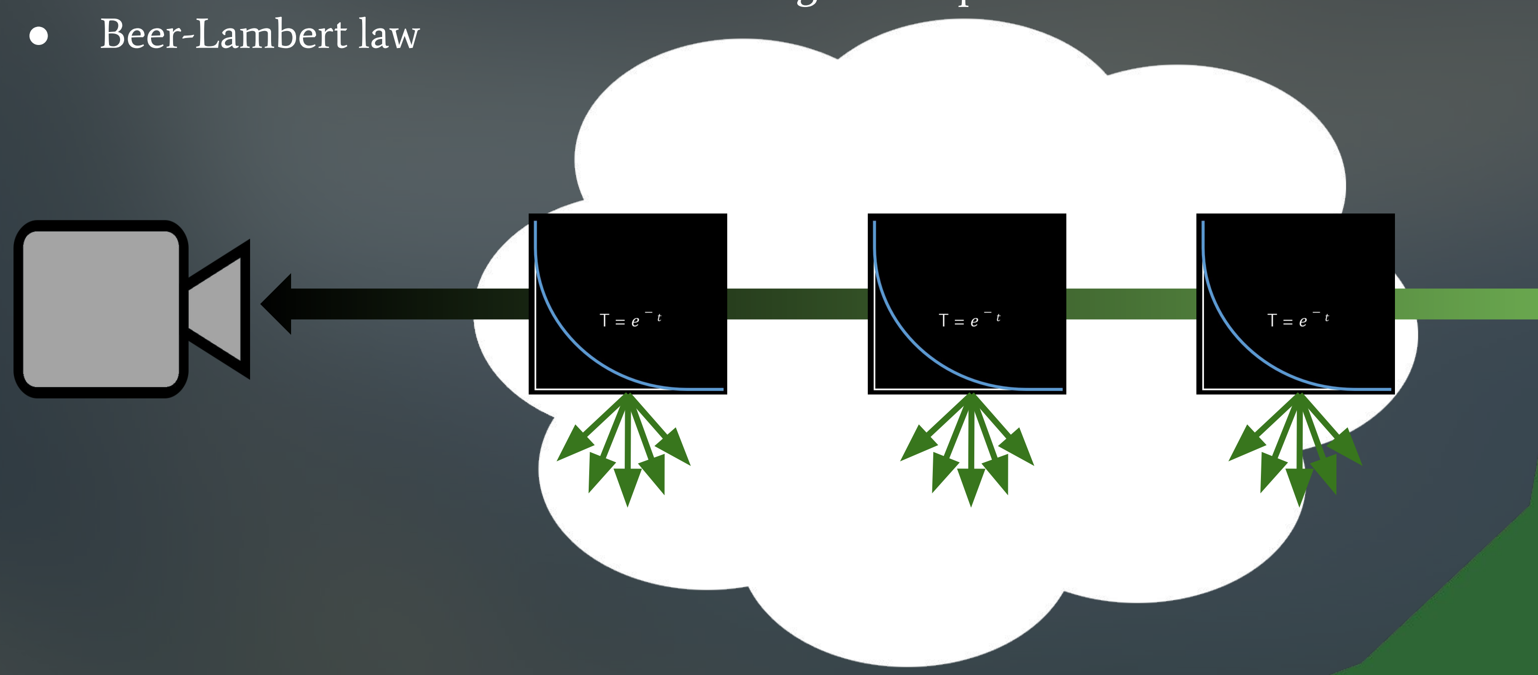
Lighting : scene



Final Color = Scene Color

Lighting : out-scattering/absorption

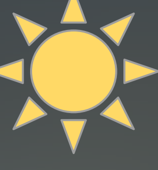
- Transmittance -> effect of out-scattering + absorption
- Beer-Lambert law

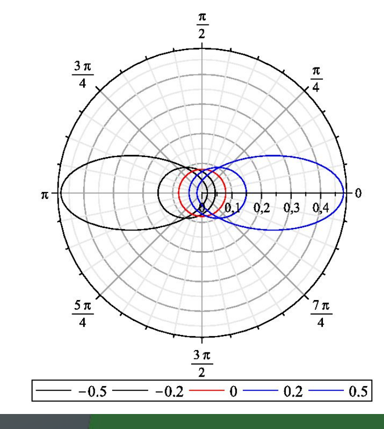


Final Color = Scene Color * Extinction Factor

Lighting : in-scattering

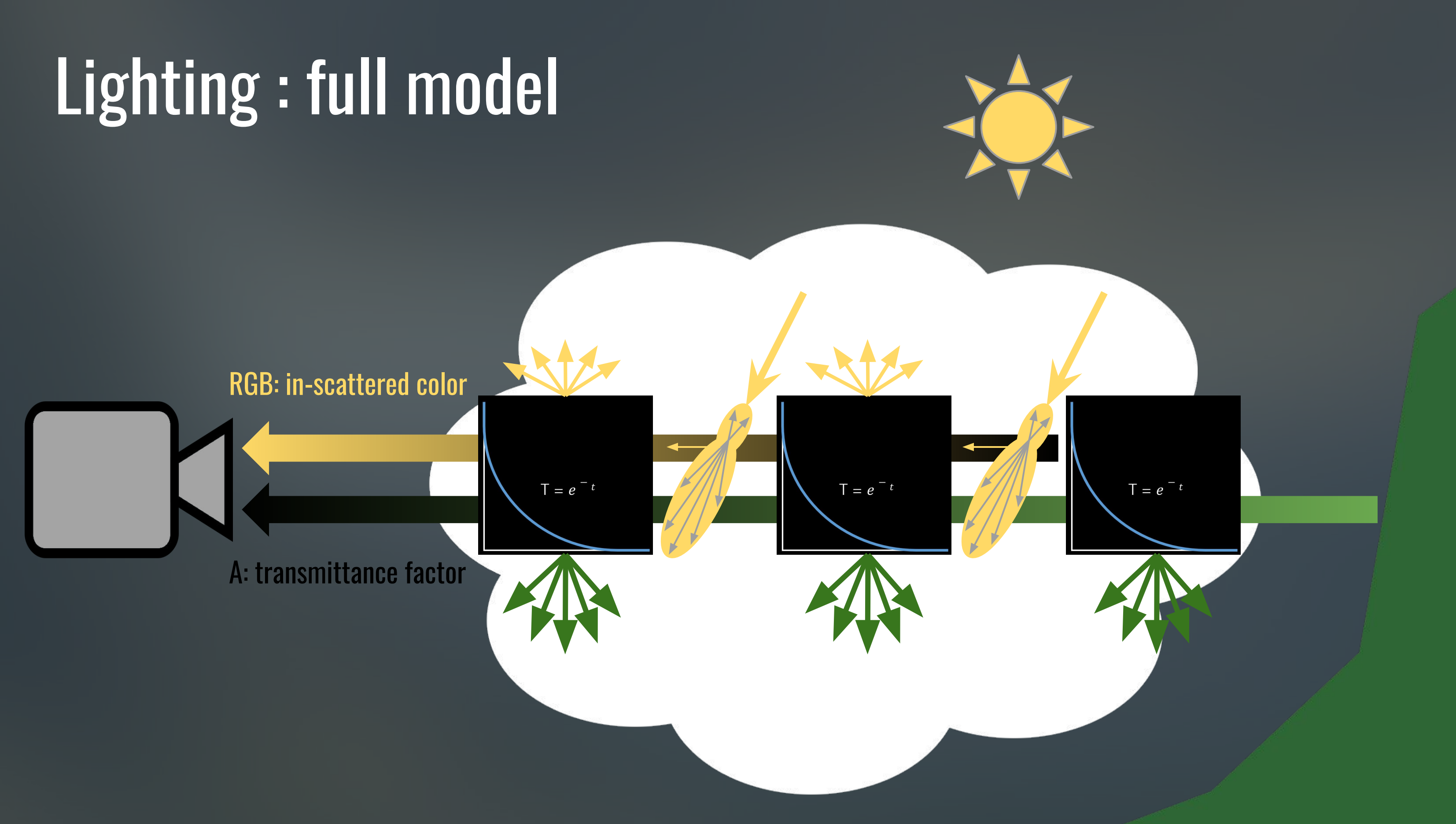
- Henyey-Greenstein phase function





Final Color = In-scattered Color

Lighting : full model



Final Color = Scene Color * Transmittance Factor + In-scattered Color

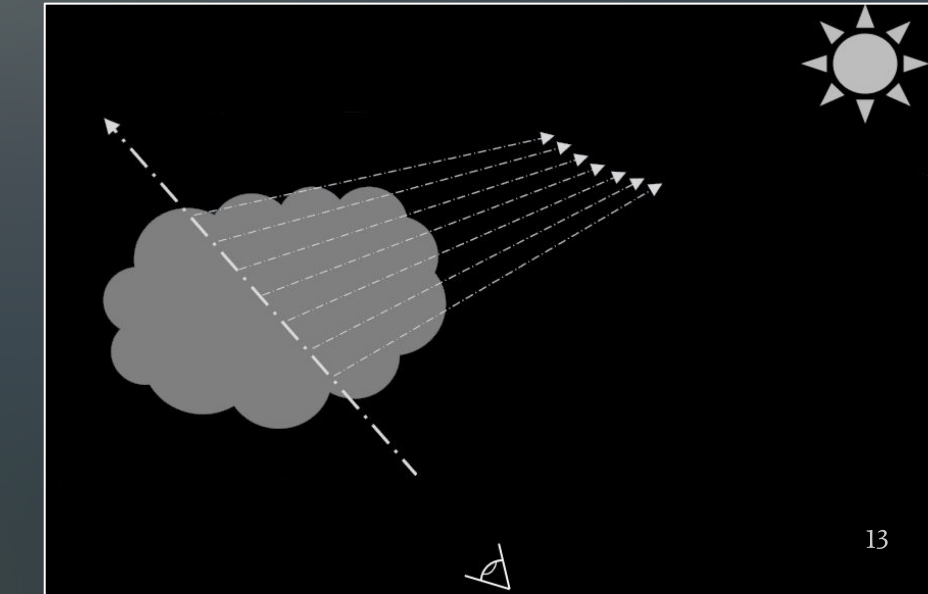
Rendering : integration

In theory : light integration

$$Li(a,b) = Li(b) * T(a,b) + \int_{x=a}^{b} HG(\theta) * V(x, L^{pos}) * T(a,x) * L^{value}$$

- *Li(a,b)* : light received at a from b
- *Li(b)* : light emitted from b
- *T(a,b)* : transmittance between a and b
- *HG(Θ)* : phase function
- *V(x, L)* : visibility from light to x

In practice : ray-marching

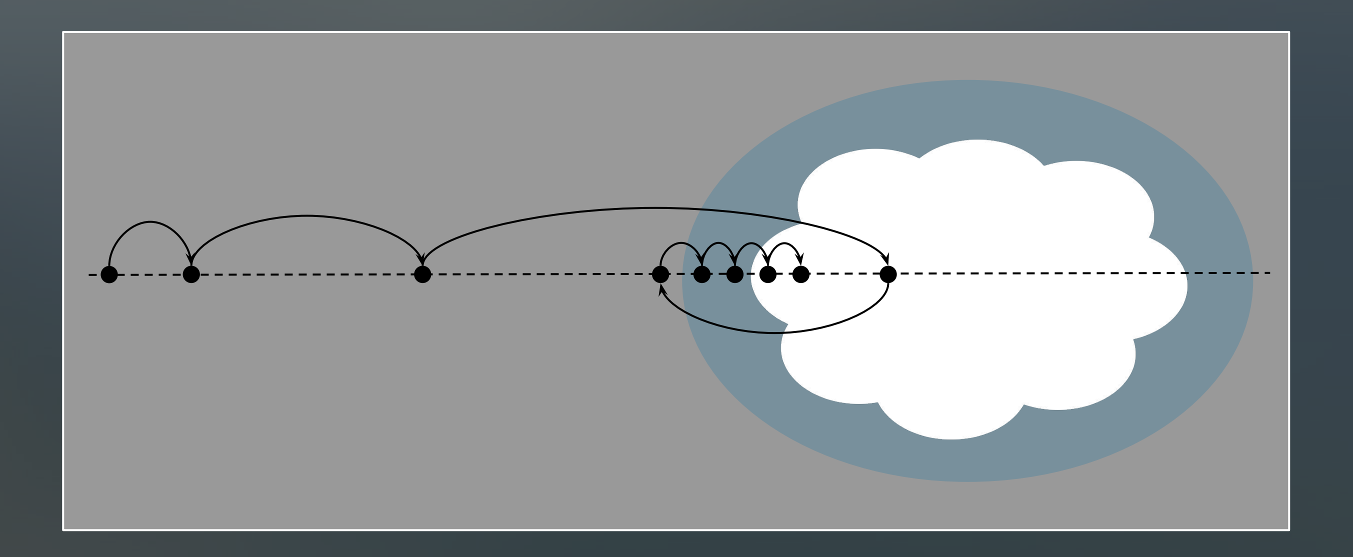





Rendering
Detailed clouds, nested ray-marching
1:1 scale scene entirely ray-marched
Render twice : camera + shadowmap
=> gtx1060 at 1080p : 60-80ms
Need order of magnitude improvement!

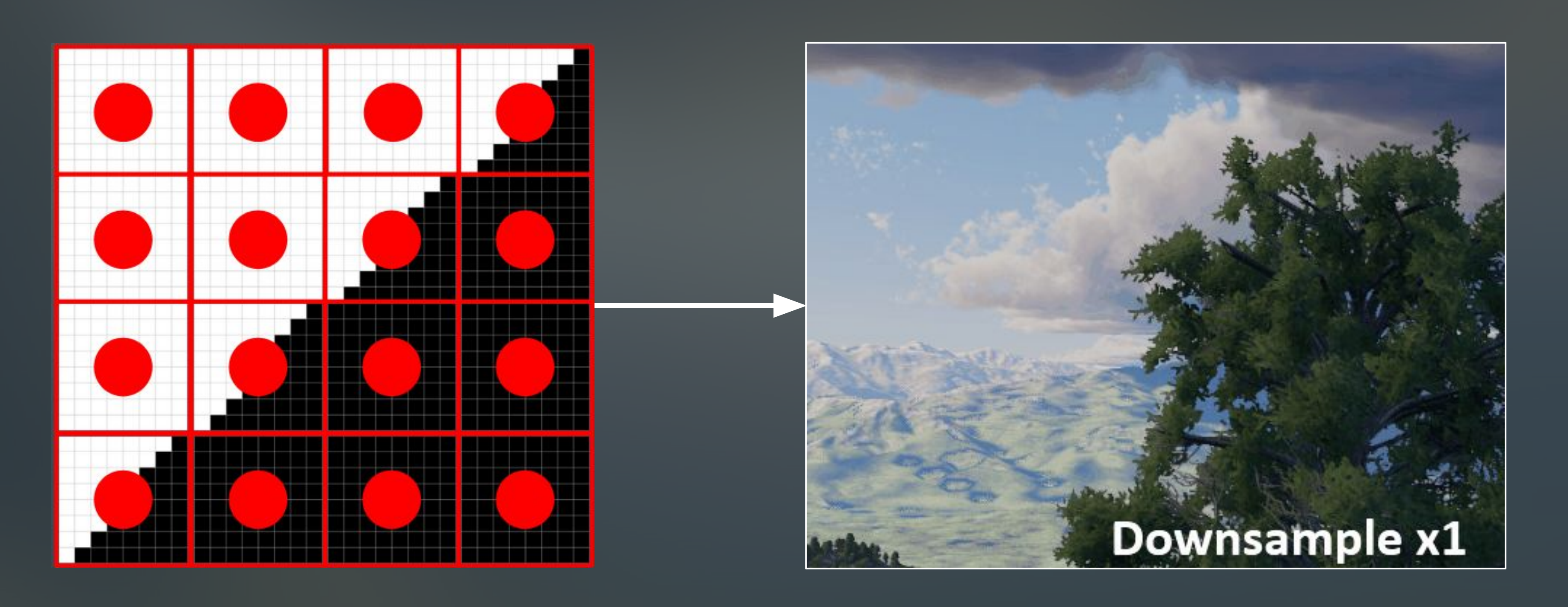
Optimizing ray-marching

- Early exit when transmittance is low
- Increase step size in empty space, step back when cloud found
- Optimize atmosphere model for ray-marching
- . . .

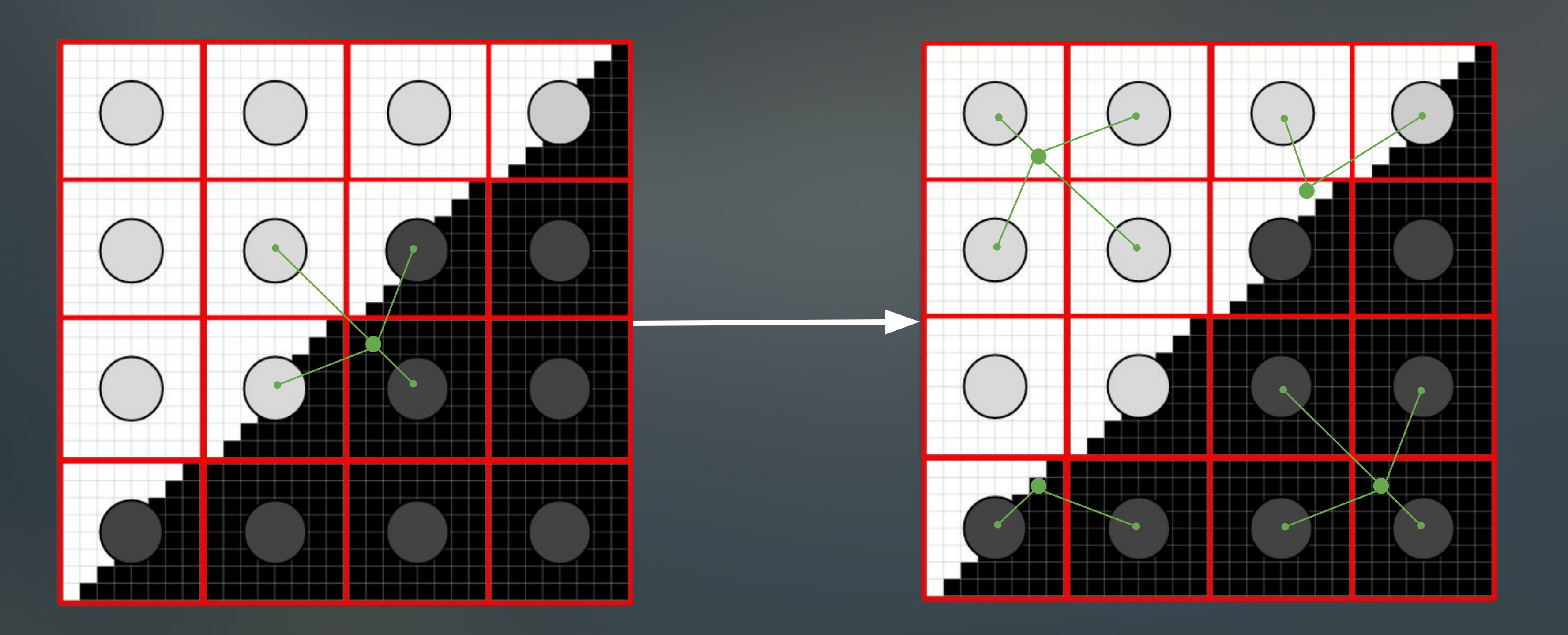


Downsampling

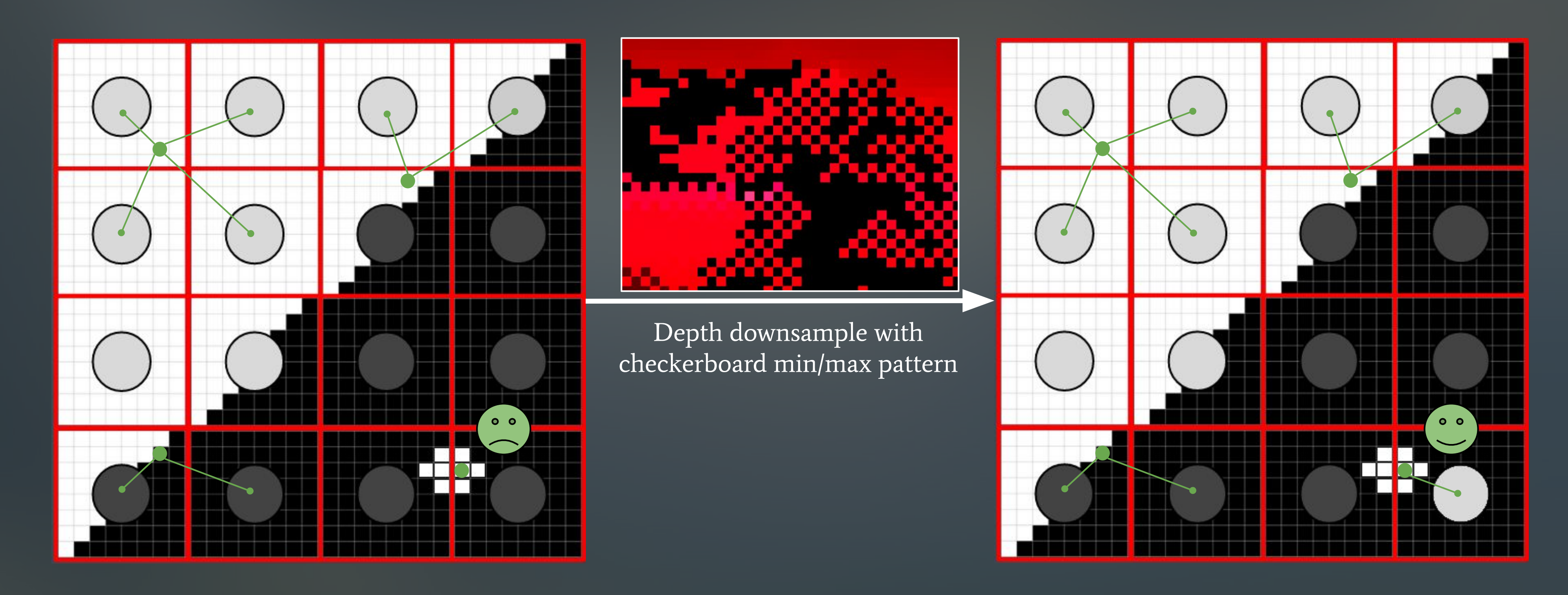
Spatial downsampling



Reconstruction filter



Reconstruction filter



Spatial downsampling





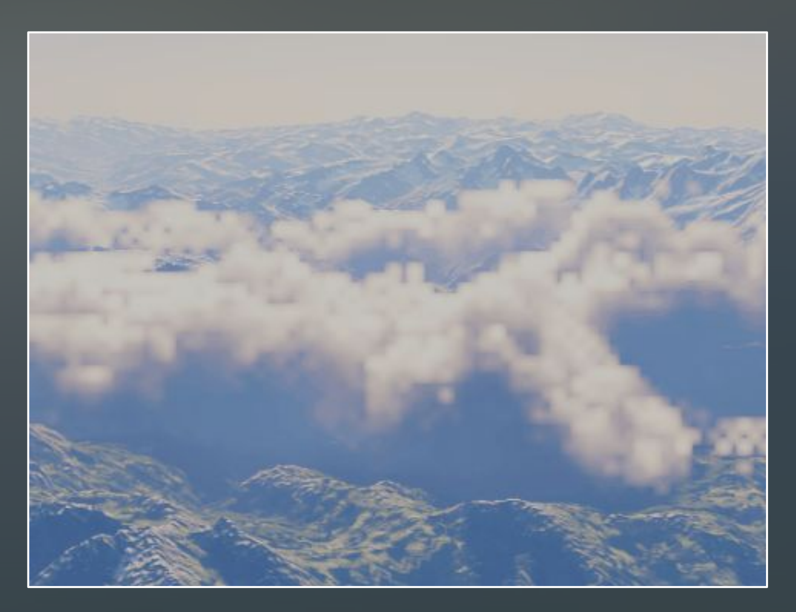
With reconstruction filter and depth min/max downsample

Spatial downsampling

Aiming for large downscaling factors...



- Many different depths in single tile



- High frequency signal in the distance

Back to basics

Sponza SSAO

Back to basics : Sponza SSAO



Reference (720p, 16k samples, RTX2080) : **272ms**

Back to basics : Sponza SSAO

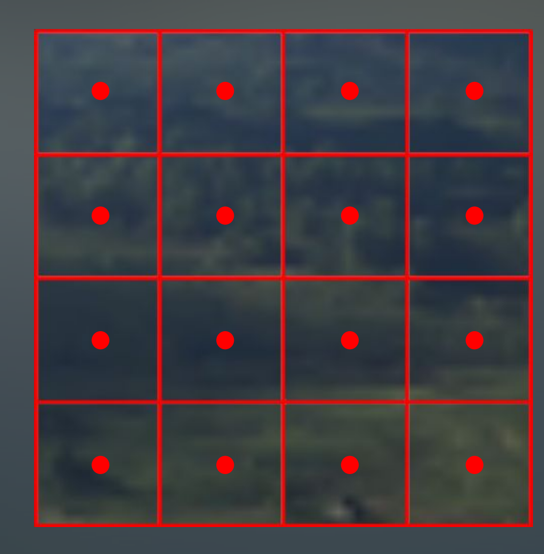


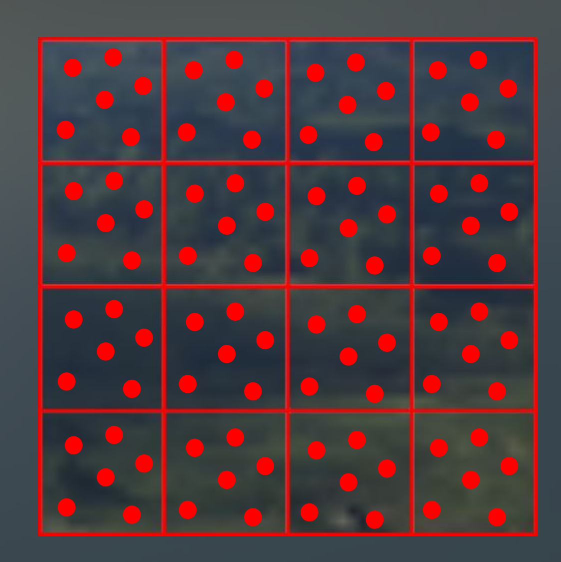
Reference : 272ms

Spatial x8 downsampling : 4ms

Temporal AA/supersampling







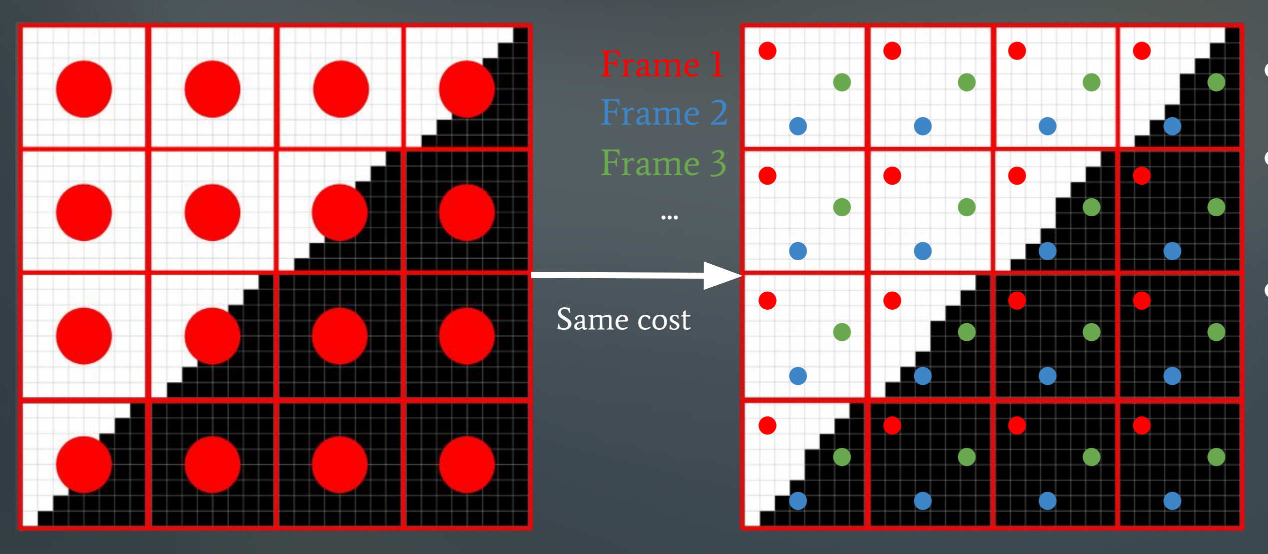
Temporal AA/supersampling

Frame 1
Frame 2
Frame 3
...

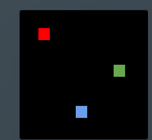
- Sub-pixel shift of projection matrix each frame
- Reproject previous frame with motion vectors
- Combine old sample with new (exponential moving average)



Temporal downsampling

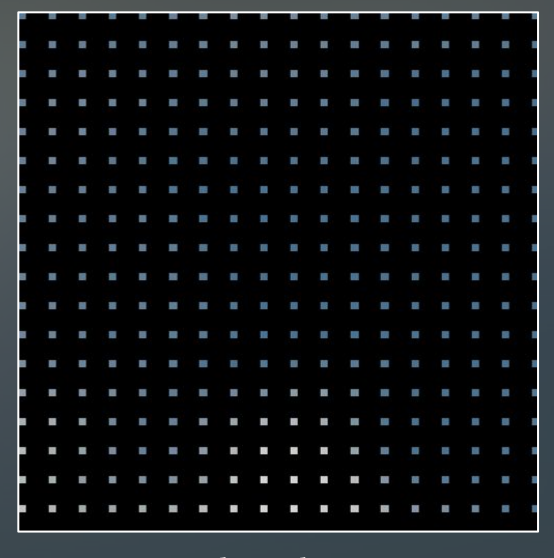


- Render full resolution buffer
- Reproject previous frame for skipped pixels
- In each tile, update only one pixel



Temporal downsampling

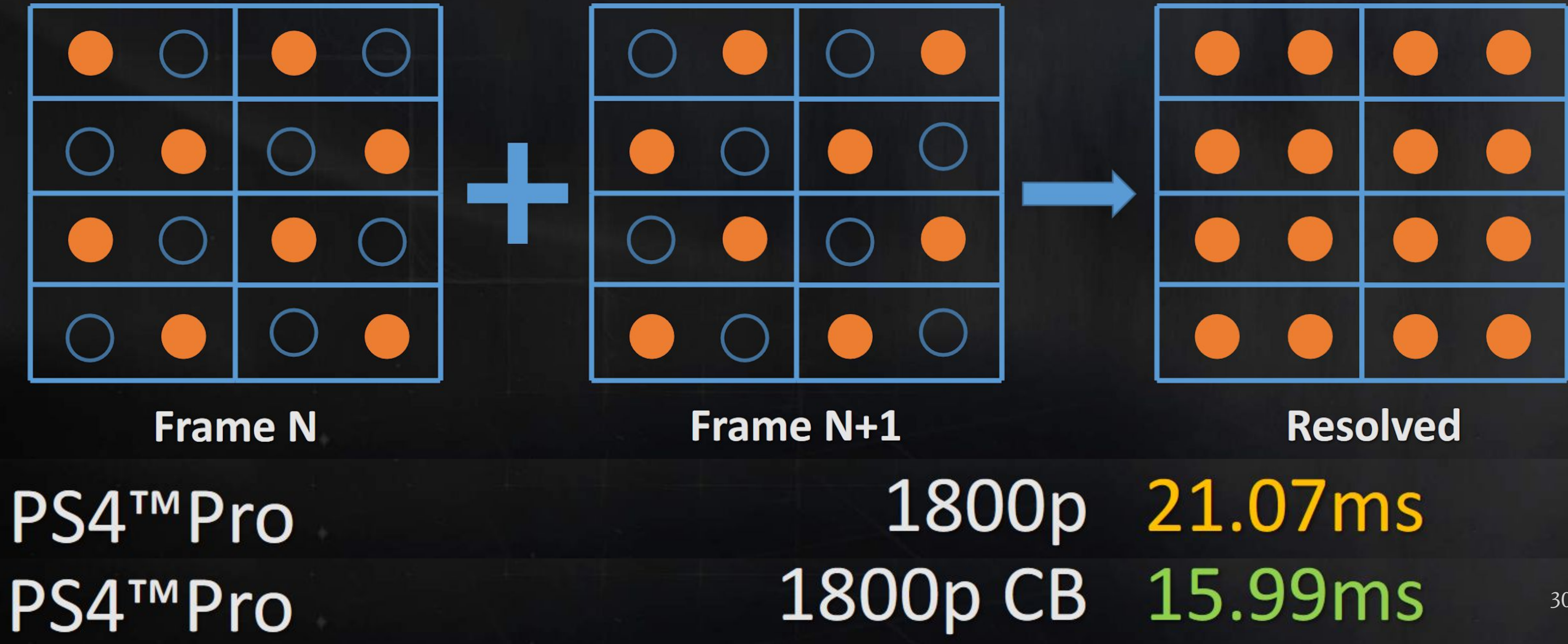
- Introduced in Horizon Zero Dawn's cloud system, reused everywhere
- 1/16 pixel ray-marched per frame
- Simple for "skybox" clouds
  → only rotational reprojection
- Build on this
  - clouds part of the scene
  - push downsampling further



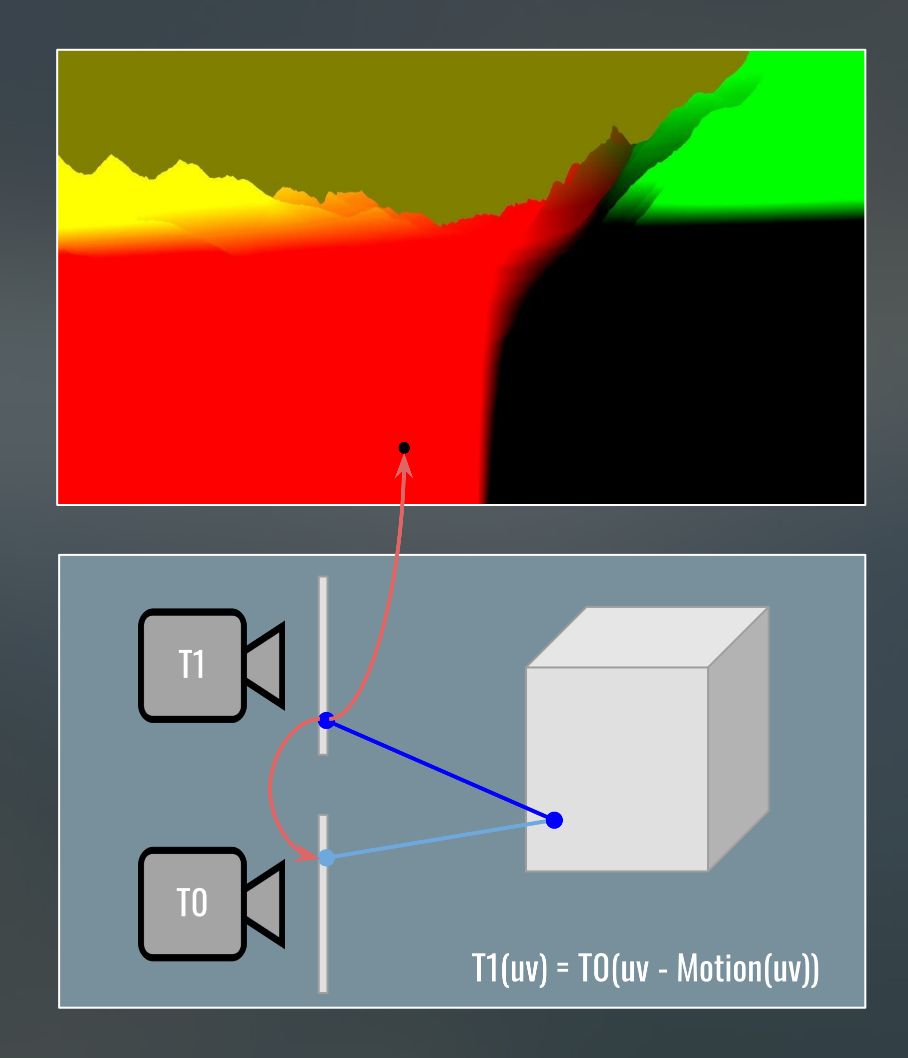
[Schneider15]

Checkerboard rendering

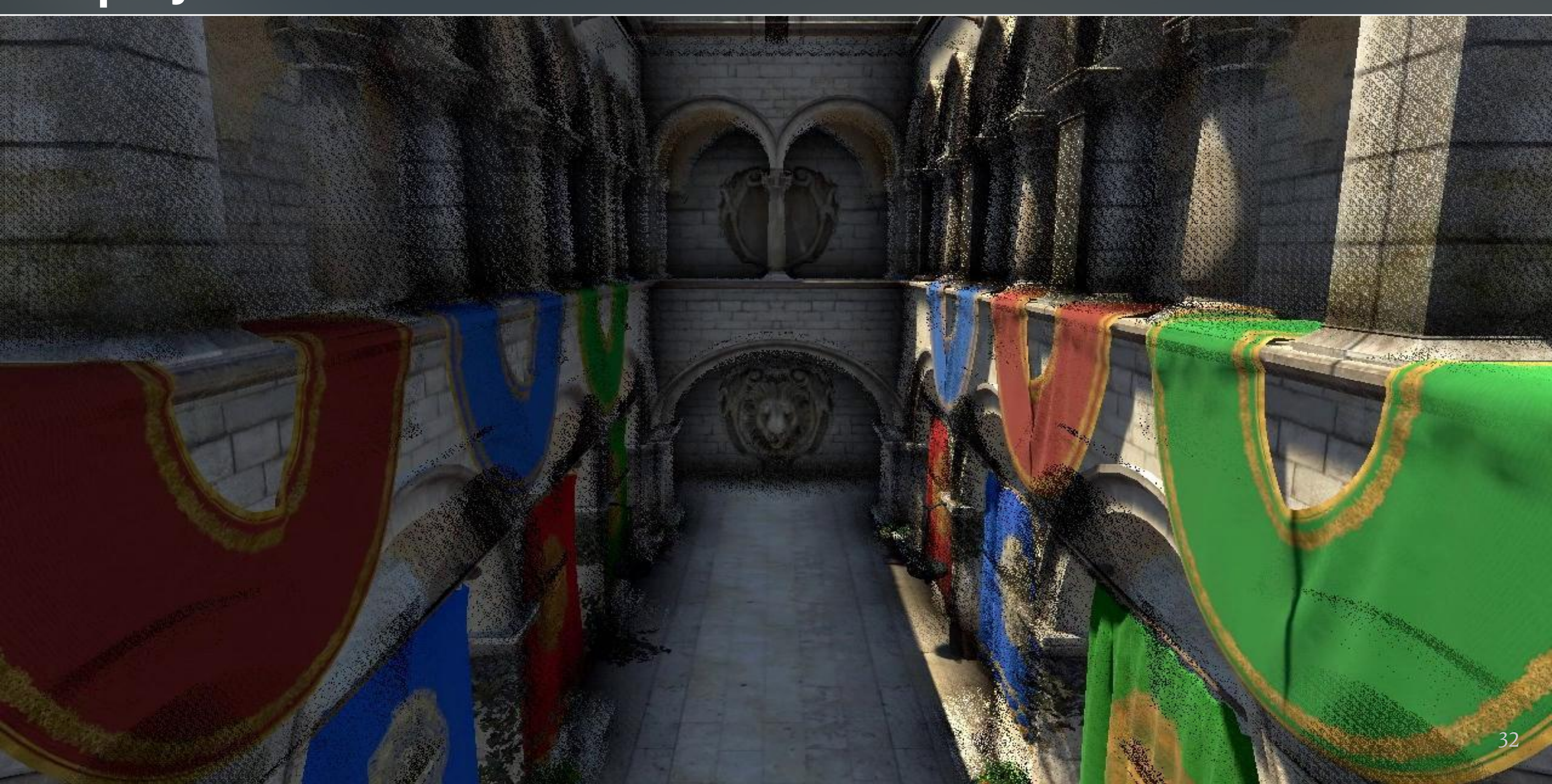
- Frostbite implementation
- Full res geometry, ½ res shading



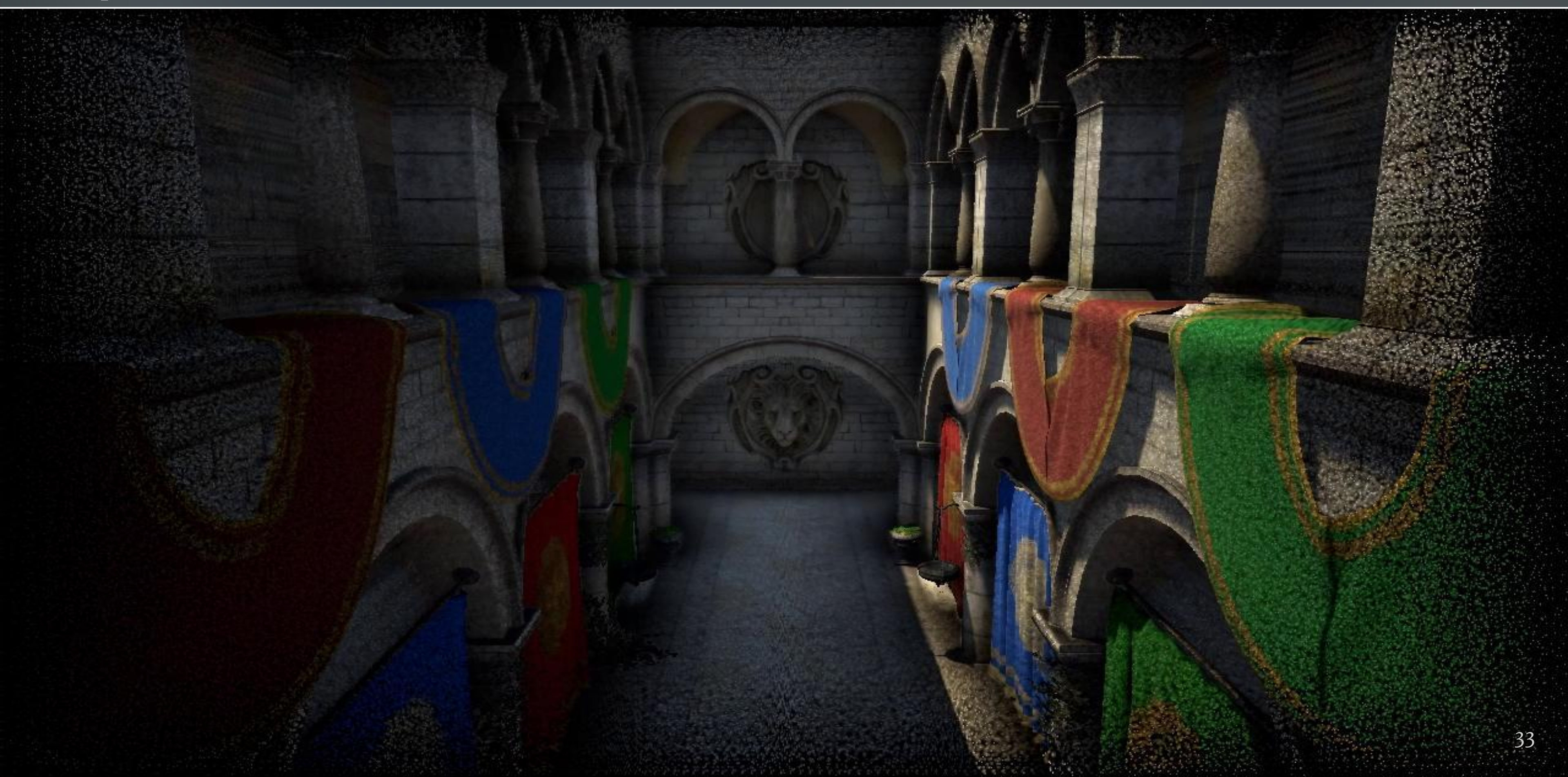
Reprojection



Reprojection



Reprojection



Reprojection



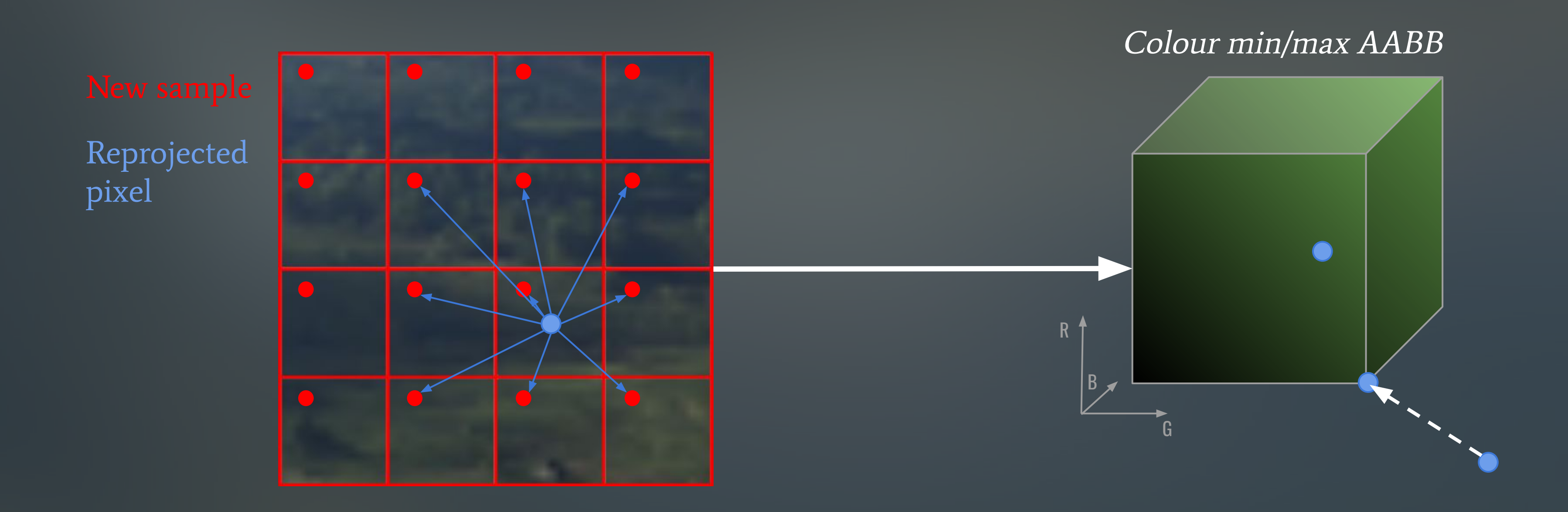
Reprojection validation/rejection

- With popularity of TAA...
- Lots of different rejection techniques, none perfect
- Often with ad-hoc parameters to tweak
- Lots of edge cases

- *Neighbourhood clamping* is behind the recent widespread use of TAA



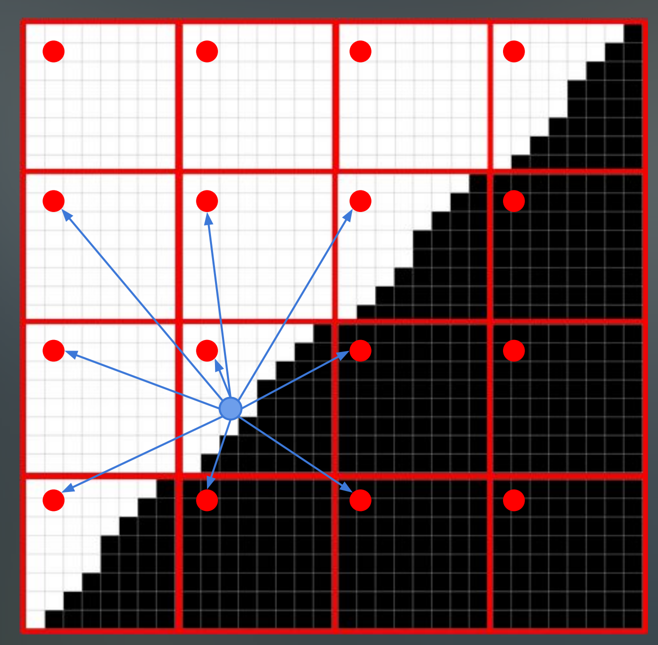
Neighbourhood clamping



Neighbourhood clamping

New sample

Reprojected sample

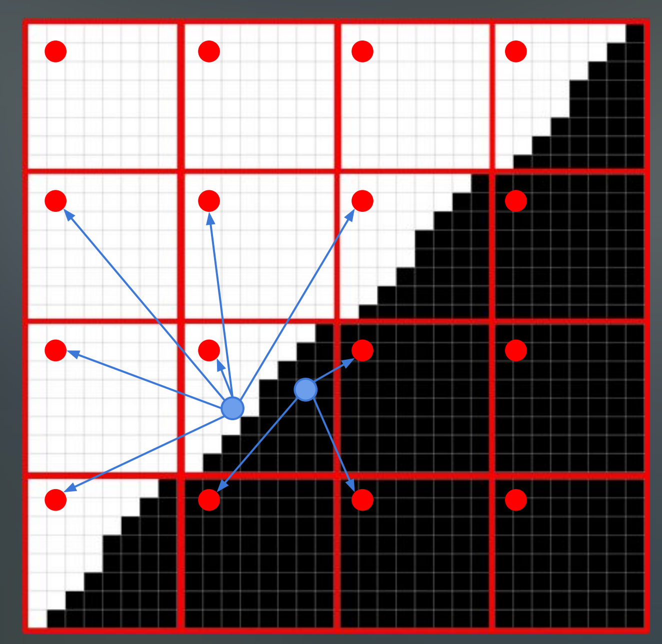


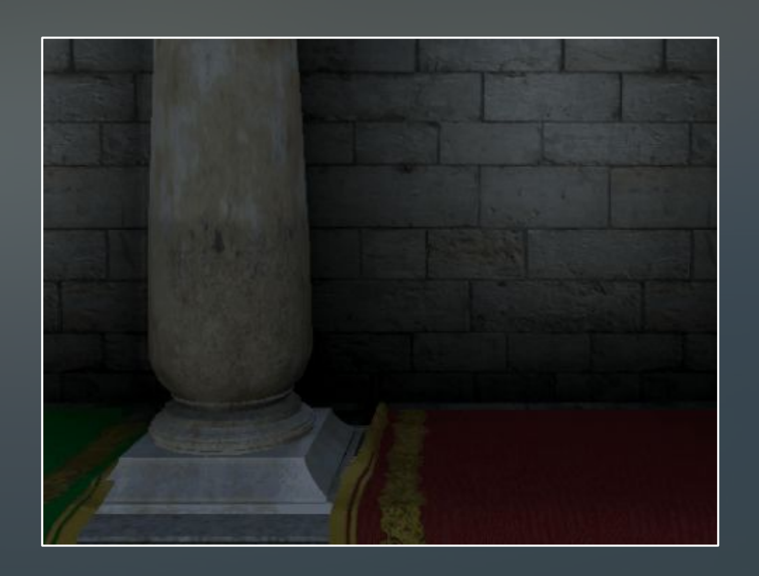


Neighbourhood clamping

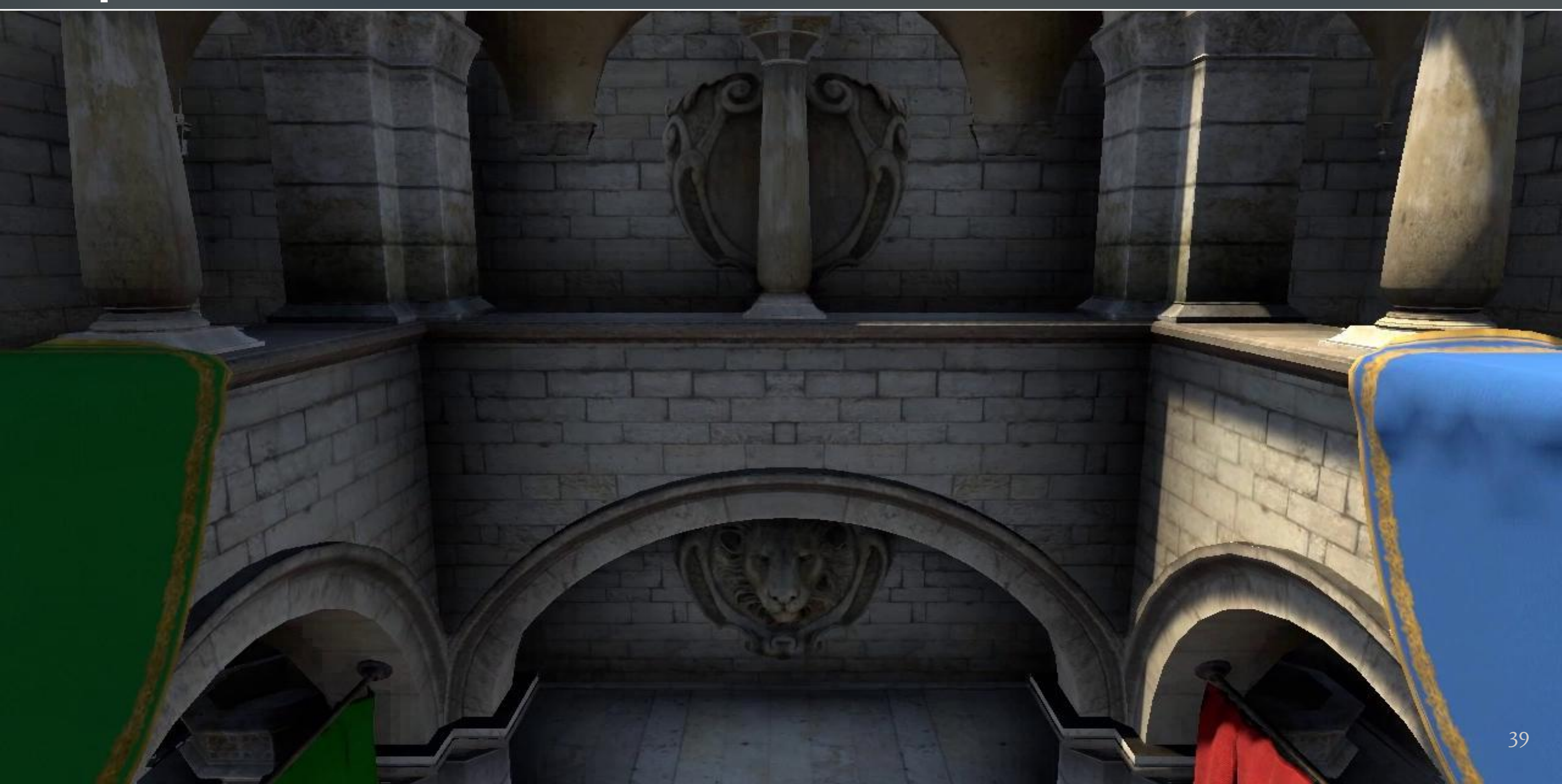
New sample

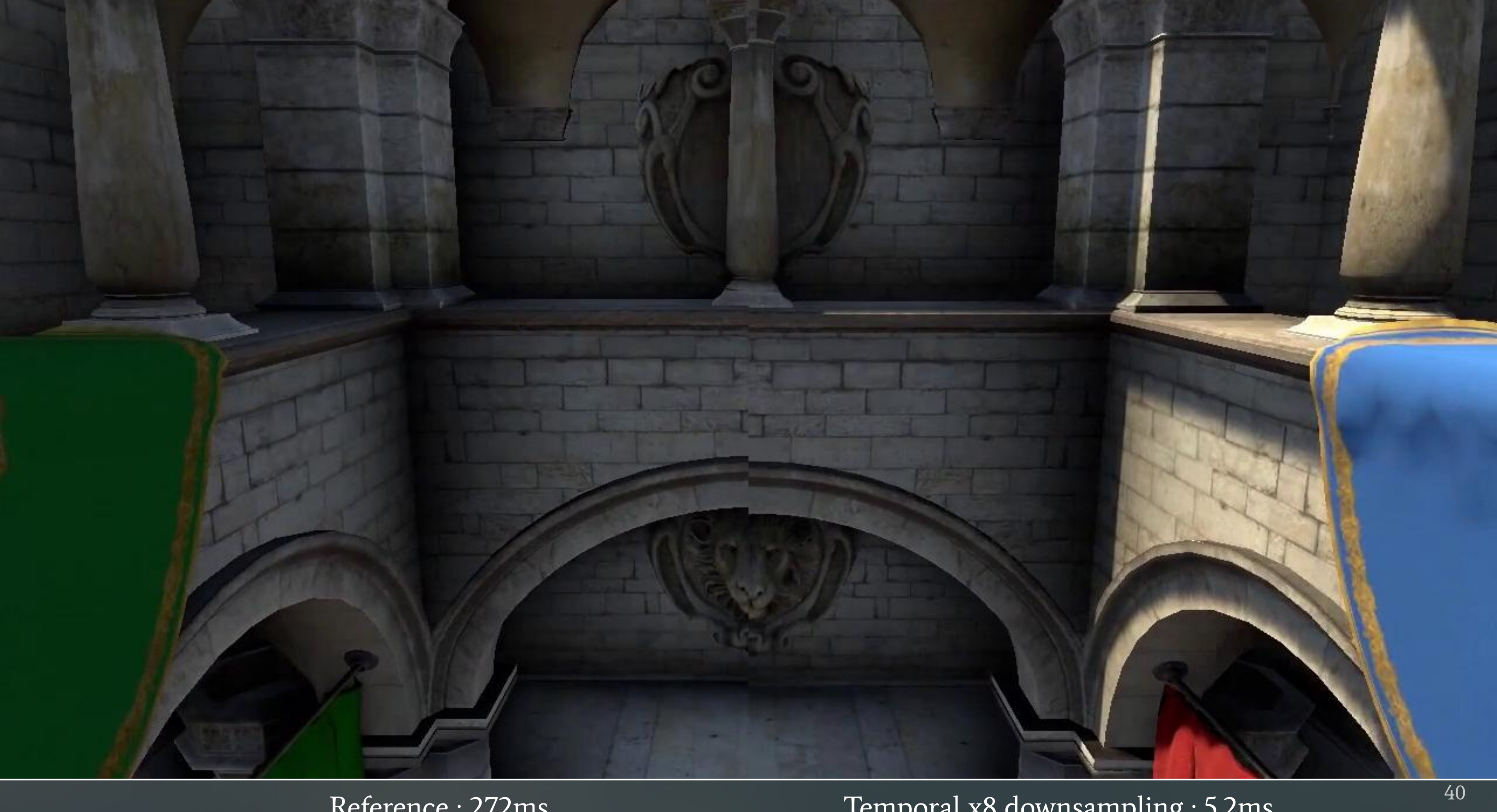
Reprojected sample





Reprojection





Reference : 272ms

Temporal x8 downsampling : 5.2ms

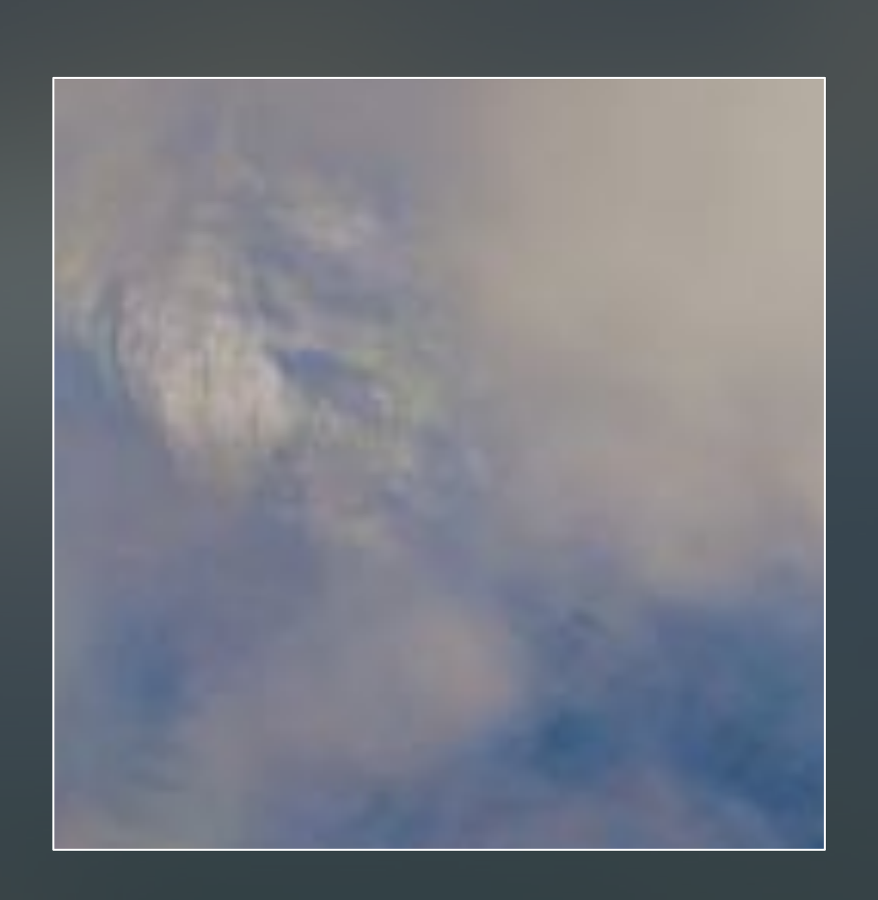
Reprojecting Clouds

Reprojecting clouds

- What depth to use ?
- Frostbite : transmittance-weighted mean cloud depth [Hillaire16]

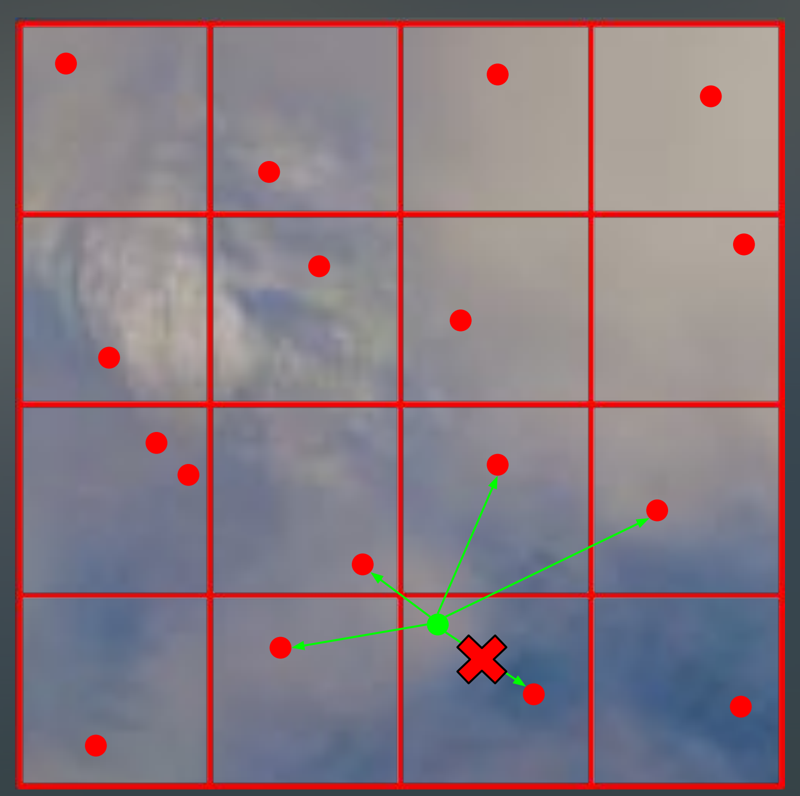
$$Depth_{cloud} = \frac{\sum_{n=0}^{N} Tr(x_n) * Depth(x_n)}{\sum_{n=0}^{N} Tr(x_n)}$$

- Basis for reprojection depth
- Hard to get right : different cases need different mean depths



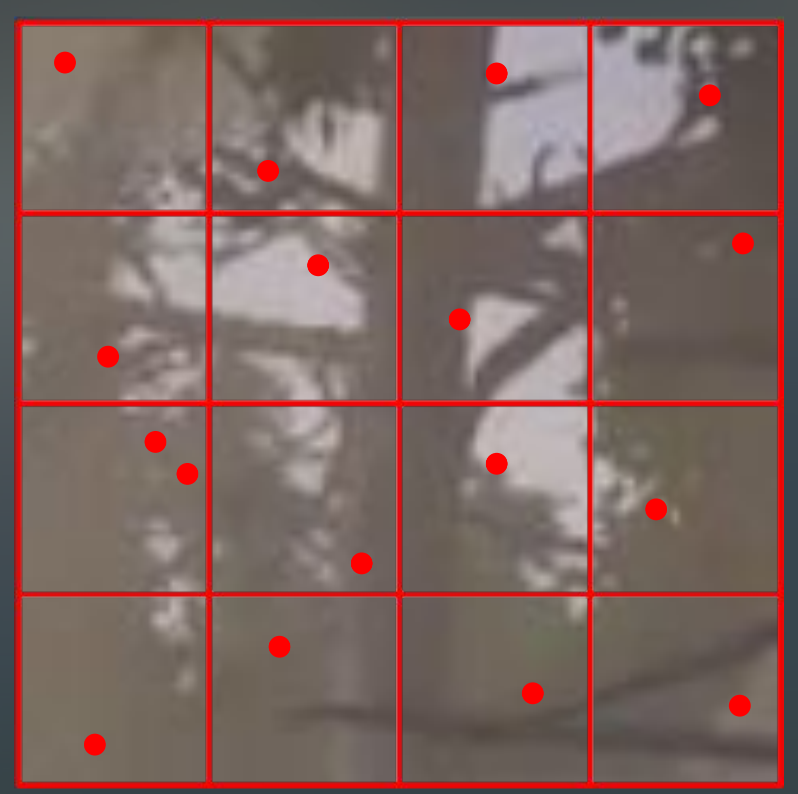
Reprojecting clouds

- Reprojection needs access to new depth
- We only have very sparse up-to-date depth data
- Can’t use old depth values : use tile’s new sample ?
- New sample of current tile might not be part of the cloud front
- Settled on average of valid depths in 3x3 neighbourhood



Reprojecting clouds

- Volumetric atmosphere result might be both uncorrelated to opaque scene depth…
- Or very correlated to it
- Need reprojection to work in all cases



Reprojecting clouds






Temporal x8 downsampling : 8ms
Reference : 70ms
1080p, gtx1060



Temporal x8 downsampling : 30ms (reference 330ms)
1080p, gtx1060

Limitations

Future Work

Reprojecting volumes

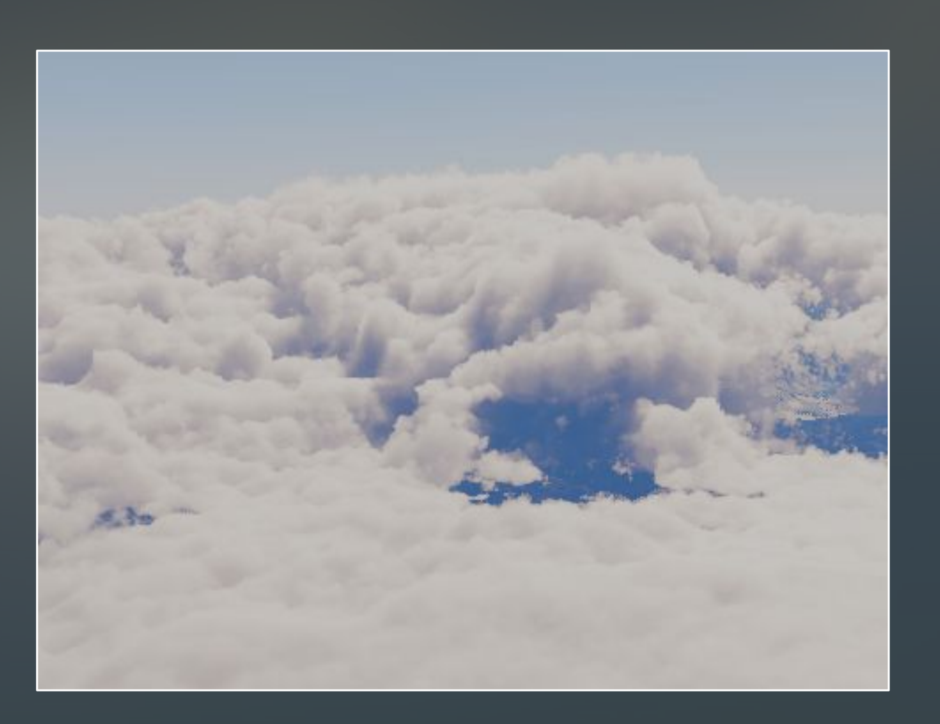
Neighbourhood clamping

- 1/64 pixel updated per frame → 1 second to converge at 60fps
- Object passing in front = reset, no valid history
- Make less visible with blur filter where clamping was aggressive



Neighbourhood clamping

- Clamping too aggressive due to missing values in min/max AABB estimation
- Modulate according to screen motion
- Hard to get right for every case
- Variance AABB clamping [Salvi2016]



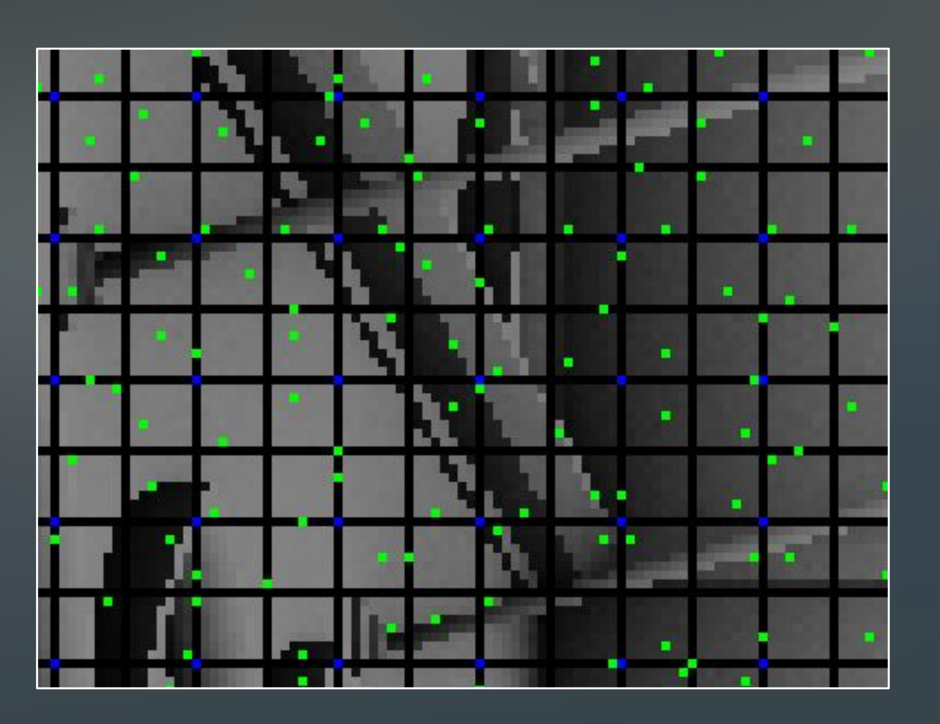
Neighbourhood clamping

- Assumes at least 1 new sample in 3x3 neighbourhood on same surface than reprojected old sample
- With large 8x8 tiles... assumption wrong in lots of cases



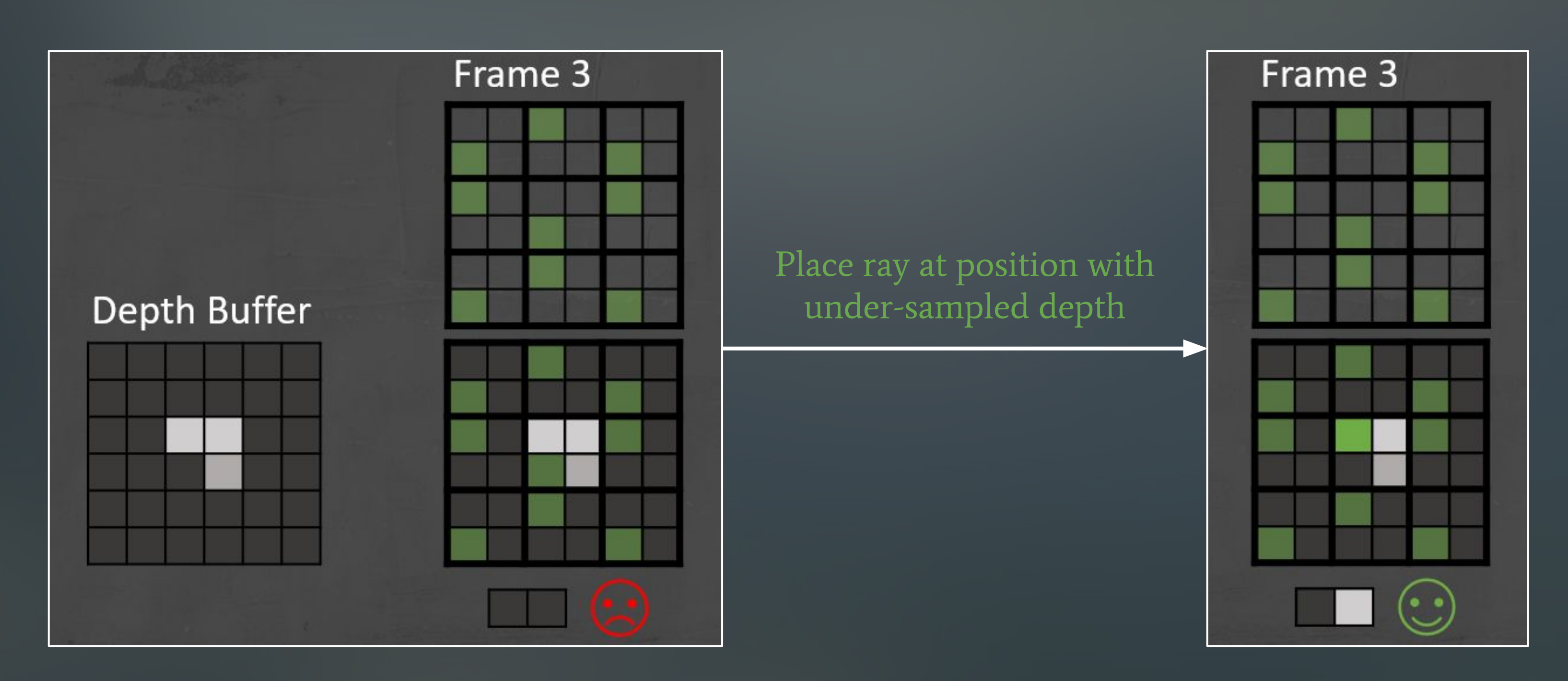
Neighbourhood clamping

- Assumes at least 1 new sample in 3x3 neighbourhood on same surface than reprojected old sample
- With large 8x8 tiles... assumption wrong in lots of cases



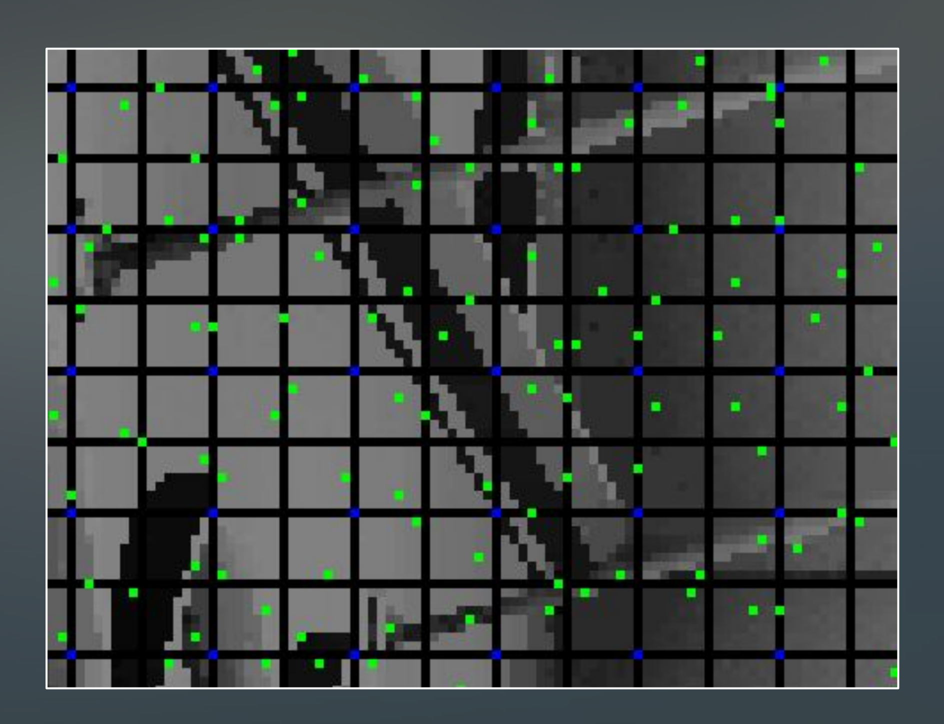
Neighbourhood clamping

- Red Ded Redemption 2 : 2x2 temporal tiles, manually adjust sample position before ray-marching in edge-cases



Neighbourhood clamping

- Experimented same idea with 8x8 tiles
- Concept of re-targeting samples where needed promising :
  → more samples where signal is high frequency (distant clouds)




Thanks